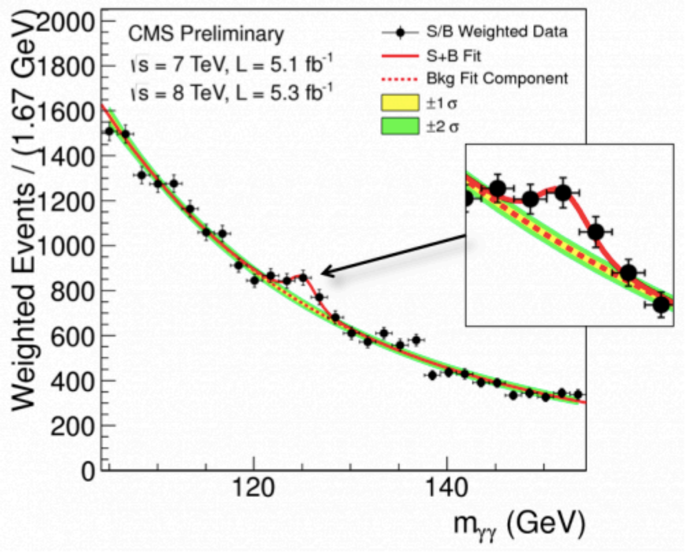


The basis of the estimation of the *local* (*) *statistical significance* of a physics signal

A.Pompili - SDAL course - Exercise 11

(*) "local" implies that the physical signal is ***already known*** and we are ***confirming*** it;
in the case of presence of a new physical signal we need to consider the Look-Elsewhere-Effect and
we would have to compute a "global" statistical significance which requires more effort (beyond the scope of this course)

PART-1

A.Pompili - SDAL course - Exercise 11

Bibliography: **G. Cowan, K. Cranmer, E. Gross, O. Vitells, Eur. Phys. J. C 71 (2011) 1554**
- https://link.springer.com/article/10.1140/epjc/s10052-011-1554-0

A. Di Florio and A. Pompili, PoS (CHARM2020) 002
- https://inspirehep.net/files/9283f5c2a80f98076e90d2c7c0ae3c98

Profile Likelihood Ratio & Test statistic

$\boldsymbol{\mu}$: signal strength of hypothesized signal; it can be considered properly a ***signal yield*** only when μ is constrained to be $\mu \geq 0$.

To test a hypothesized value of μ ... we consider the **P**rofile **L**ikelihood **R**atio (here $\vec{\theta}$ represents a set of *nuisance parameters*):

$$\lambda(\mu) = \frac{\mathcal{L}(\mu, \hat{\hat{\vec{\theta}}})}{\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})} \quad \left[\frac{\ldots where\ \hat{\hat{\vec{\theta}}}\ are\ the\ values\ of\ \vec{\theta}\ that\ maximize\ \mathcal{L}\ for\ a\ specified\ \mu}{\ldots where\ \hat{\mu}\ and\ \hat{\vec{\theta}}\ are\ the\ values\ maximizing\ the\ Likelihood\ function\ \mathcal{L}}\right]$$

Profile Likelihood Function

Intuitively it measures the level of *agreement* between data and the hypothesized value of μ : $\lambda(\mu) = \frac{\mathcal{L}(\mu)}{\mathcal{L}(\hat{\mu})} \Rightarrow \begin{cases} \text{low } \lambda \Rightarrow \textit{poor} \\ \text{high } \lambda \Rightarrow \textit{good} \end{cases}$

Profile Likelihood Ratio & Test statistic

$\boldsymbol{\mu}$: signal strength of hypothesized signal; it can be considered properly a ***signal yield*** only when μ is constrained to be $\mu \geq 0$.

To test a hypothesized value of μ ... we consider the Profile Likelihood Ratio (here $\vec{\theta}$ represents a set of *nuisance parameters*):

$$\boldsymbol{\lambda(\mu)} = \frac{\mathcal{L}(\mu, \hat{\hat{\vec{\theta}}})}{\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})} \left[\frac{\text{... where } \hat{\hat{\vec{\theta}}} \text{ are the values of } \vec{\theta} \text{ that maximize } \mathcal{L} \text{ for a specified } \mu}{\text{... where } \hat{\mu} \text{ and } \hat{\vec{\theta}} \text{ are the values maximizing the Likelihood function } \mathcal{L}} \right]$$

$\mathcal{L}(\mu, \hat{\hat{\vec{\theta}}})$ → Profile Likelihood Function

Intuitively it measures the level of *agreement* between data and the hypothesized value of μ : $\lambda(\mu) = \frac{\mathcal{L}(\mu)}{\mathcal{L}(\hat{\mu})} \Rightarrow \begin{cases} \text{low } \lambda \Rightarrow \textit{poor} \\ \text{high } \lambda \Rightarrow \textit{good} \end{cases}$

Test statistic : $t_\mu = -2 ln\lambda(\mu) = -2\left[ln\mathcal{L}(\mu, \hat{\hat{\vec{\theta}}}) - ln\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})\right] = 2\left[\underbrace{-ln\mathcal{L}\left(\mu, \hat{\hat{\vec{\theta}}}\right)}_{NLL0} - \left(\underbrace{-ln\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})}_{NLL1}\right)\right] \equiv 2 \cdot \Delta NLL$

$\Delta NLL \equiv NLL0 - NLL1$

where $NLL0$($NLL1$) indicates the **N**eg-**L**og-**L**ikelihood associated to the null-0 (alternative-1) hypotheses

This test statistic can be used for a test of $\mu = 0$ for purposes of establishing the existence of a signal process.

In the case of $\mu = 0$, $NLL0$($NLL1$) indicates the bkg−only(sig+bkg) hypothesis.
The sig+bkg hypothesis can be represented by either $\mu \geq 0$ or $\mu \neq 0$. We will consider both cases in the following.

Note: $\boldsymbol{\lambda(\mu)}$ has the important advantage that - for a sufficiently large event sample - its distribution approaches a ...
... well defined form (according to Wilks' Theorem) [see later]. This is true also in presence of adjustable nuisance parameters.

p-value & Statistical Significance - I

In general, by denoting with f the p.d.f. of the test statistic t_μ,
the p-value of the hypothesized value of μ for an observed t_μ (denoted as $t_{\mu,obs}$) can be expressed as:

$$p_\mu = \int_{t_{\mu,obs}}^{\infty} f(t_\mu|\mu)dt_\mu$$

← value of the test statistic observed in the data

From: **EPJ C 71 (2011) 1554**

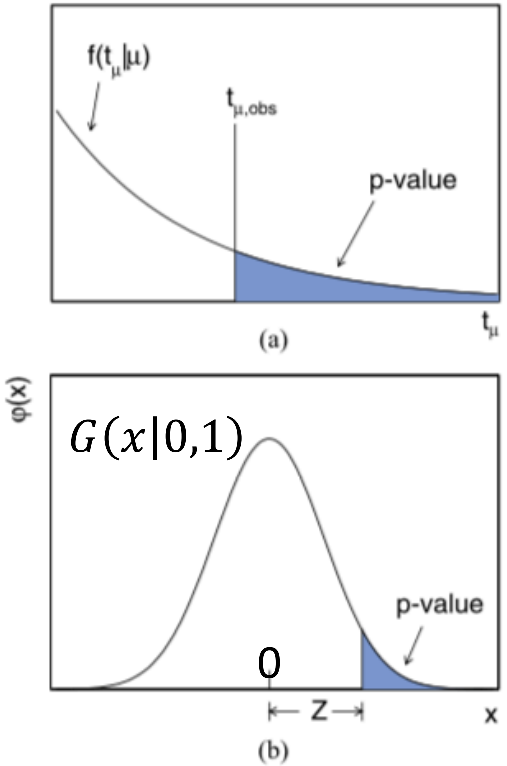


Fig. 1 **(a)** Illustration of the relation between the p-value obtained from an observed value of the test statistic t_μ. **(b)** The standard normal distribution $\varphi(x) = (1/\sqrt{2\pi})\exp(-x^2/2)$ showing the relation between the significance Z and the p-value

In HEP we usually convert the p-value into an equivalent significance (Z), defined such that a Gaussian distributed variable x found Z standard deviations above its null mean has an upper-tail probability equal to p (the one-sided definition is used here as it gives $Z = 0$ for $p = 0.5$).
Once introduced the cumulative distribution Φ (c.d.f.) of the Standard Gaussian one has:

$$\Phi(Z) = \int_{-\infty}^{Z} G(x|0,1)dx = 1 - \int_{Z}^{\infty} G(x|0,1)dx = 1 - p_Z$$

…thus, the following expression for Z can be derived: $Z = \Phi^{-1}(1 - p_Z)$

…where Φ^{-1} is the inverse of the c.d.f and is called *quantile* of the Standard Gaussian.

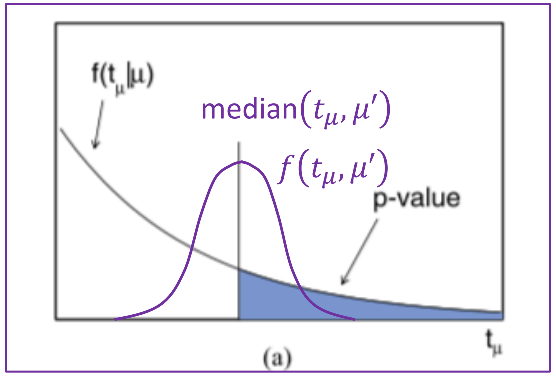
In HEP the observation/discovery requires at least $Z = 5$ namely $p = 2.87 \cdot 10^{-7}$;

Viceversa to exclude a signal hypothesis at 95% C.L. , $p = 0.05$ corresponds to $Z = 1.64$.

p-value & Statistical Significance - II

In the case of a single parameter of interest (the strength parameter μ) it is possible to find an approximate distribution for the profile likelihood ratio.
Consider a test of μ which can be either $\mu = 0$ (for discovery) or $\mu \neq 0$ (for upper limit) and suppose the data are distributed according to a strength parameter μ'.

Wald (1943) showed that $-2ln\lambda(\mu) = \frac{(\mu-\hat{\mu})^2}{\sigma^2} + \mathrm{O}(1/\sqrt{N})$ [known as *Wald approximation*]



… where $\hat{\mu}$ follows a Gaussian distribution with mean μ' & standard deviation σ
[i.e. $\mathrm{E}[\hat{\mu}] = \mu'$, σ derived from Cov. Matrix] and N represents the data sample size.

Generally, this is introduced to quantify how sensitive we are to a potential discovery, e.g. by a given median significance assuming some nonzero strength parameter μ'.

In the large limit sample ($N \to \infty$) we can neglect the $\mathrm{O}(1/\sqrt{N})$ term: $\boldsymbol{t_\mu} = -2ln\lambda(\mu) \cong \frac{(\mu-\hat{\mu})^2}{\sigma^2}$

… and one can show that $\boldsymbol{t_\mu}$ follows a non-central χ^2_1 distribution, with the non-centrality term being $\Lambda = \frac{(\mu-\mu')^2}{\sigma^2}$:

$$\boldsymbol{f(t_\mu, \Lambda)} \cong \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{2\sqrt{\boldsymbol{t_\mu}}} \cdot \left[e^{-1/2(\sqrt{\boldsymbol{t_\mu}}+\sqrt{\Lambda})^2} + e^{-1/2(\sqrt{\boldsymbol{t_\mu}}-\sqrt{\Lambda})^2}\right]$$

p-value & Statistical Significance - III

In the special case $\mu = \mu'$ (and thus $\Lambda = 0$) … $\boldsymbol{t_\mu}$ follows a central χ_1^2 distribution [a result shown earlier by Wilks (1938)] :

$$\boldsymbol{f(t_\mu|\mu)} \cong \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{2\sqrt{\boldsymbol{t_\mu}}} \cdot \left[e^{-1/2\left(\sqrt{\boldsymbol{t_\mu}}\right)^2} + e^{-1/2\left(\sqrt{\boldsymbol{t_\mu}}\right)^2}\right] = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{2\sqrt{\boldsymbol{t_\mu}}} \cdot 2e^{-1/2t_\mu} = \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{\boldsymbol{t_\mu}}} \cdot e^{-\frac{1}{2}t_\mu} \equiv \frac{1}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{\boldsymbol{t_\mu}}} \cdot e^{-\frac{1}{2}\left(\sqrt{\boldsymbol{t_\mu}}\right)^2}$$

It can be checked […] that the cumulative distribution of $\boldsymbol{t_\mu}$ is: $\boldsymbol{F(t_\mu|\mu)} \cong 2\Phi\left(\sqrt{t_\mu}\right) - 1$ where Φ is the c.d.f. of the standard Gaussian

The p-value of a hypothetical value of μ for an observed value of $\boldsymbol{t_\mu}$ is:

$$p_\mu = \int_{t_{\mu,obs}}^{\infty} f(t_\mu|\mu)dt_\mu = 1 - \int_{-\infty}^{t_{\mu,obs}} f(t_\mu|\mu)dt_\mu = 1 - F(t_\mu|\mu) \cong 2\left(1 - \Phi\left(\sqrt{t_\mu}\right)\right)$$

… and the significance corresponding to the p-value is, by rewriting the expression $Z = \Phi^{-1}(1 - p_Z)$:

$$\boldsymbol{Z_\mu} = \Phi^{-1}(1 - p_\mu) = \Phi^{-1}\left(1 - 2\left(1 - \Phi\left(\sqrt{t_\mu}\right)\right)\right) = \Phi^{-1}\left(1 - 2 + 2\Phi\left(\sqrt{t_\mu}\right)\right) = \boldsymbol{\Phi^{-1}\left(2\Phi\left(\sqrt{t_\mu}\right) - 1\right)}$$ … where… $\boldsymbol{t_\mu} = -2ln\lambda(\mu)$

…that can be further simplified: $= \Phi^{-1}\left(2\Phi\left(\sqrt{t_\mu}\right) - 2\cdot\frac{1}{2}\right) = \Phi^{-1}\left(2\Phi\left(\sqrt{t_\mu}\right) - 2\Phi(0)\right) = 2\cdot\left(\Phi^{-1}\Phi\left(\sqrt{t_\mu}\right) - \Phi^{-1}\Phi(0)\right) = 2\left(\sqrt{t_\mu} - 0\right) \equiv 2\sqrt{t_\mu}$

A new test statistic for $\mu \geq 0$

We often **assume** that the presence of a signal can **only increase** the mean event rate beyond what is expected from bkg alone, namely $\mu \geq 0$. To take this into account we need to introduce a new test statistic denoted as $\tilde{t}_\mu$.
For a model where $\mu \geq 0$, if one finds data such that $\hat{\mu} < 0$ ($\hat{\mu}$ is the effective estimator, that can be negative), then the best level of agreement between data and any physical value of occurs for $\mu = 0$. Thus, the new test statistic is defined as follows:

$$\tilde{t}_\mu = -2ln\tilde{\lambda}(\mu) = \begin{cases} -2ln\dfrac{\mathcal{L}(\mu,\hat{\hat{\vec{\theta}}}(\mu))}{\mathcal{L}(\hat{\mu},\hat{\vec{\theta}})} \quad , \quad \hat{\mu} \geq 0 \\ -2ln\dfrac{\mathcal{L}(\mu,\hat{\hat{\vec{\theta}}}(\mu))}{\mathcal{L}(0,\hat{\hat{\vec{\theta}}}(0))} \quad , \quad \hat{\mu} < 0 \end{cases}$$

Again (as done with t_μ) we can quantify the level of disagreement between data & the hypothesized value of μ with the p-value … $p_\mu = \int_{\tilde{t}_{\mu,obs}}^{\infty} f(\tilde{t}_\mu|\mu)d\tilde{t}_\mu$

Test statistic for discovery of a positive signal - I

An important special case of the statistic $\tilde{t}_\mu$ is used to test $\mu = 0$ in a class of model where we assume $\mu \geq 0$: rejecting the $\mu = 0$ hypothesis effectively leads to the discovery of a new signal.

For this important case the special notation $\tilde{t}_{\mu=0} \equiv \tilde{t}_0 = q_0$ is used. Using the $\tilde{t}_\mu$ definition with $\mu = 0$ we can write:

$$q_0 = \tilde{t}_0 = -2ln\tilde{\lambda}(0) = \begin{cases} -2ln\lambda(0) & , \quad \hat{\mu} \geq 0 \\ 0 & , \quad \hat{\mu} < 0 \end{cases}$$

where, of course, $\lambda(0) = \lambda(\mu = 0) = \dfrac{\mathcal{L}(0, \hat{\hat{\vec{\theta}}}(0))}{\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})}$

Test statistic for discovery of a positive signal - I

An important special case of the statistic $\tilde{t}_\mu$ is used to test $\mu = 0$ in a class of model where we assume $\mu \geq 0$: rejecting the $\mu = 0$ hypothesis effectively leads to the discovery of a new signal.

For this important case the special notation $\tilde{t}_{\mu=0} \equiv \tilde{t}_0 = q_0$ is used. Using the $\tilde{t}_\mu$ definition with $\mu = 0$ we can write:

$$q_0 = \tilde{t}_0 = -2ln\tilde{\lambda}(0) = \begin{cases} -2ln\lambda(0) & , \quad \hat{\mu} \geq 0 \\ 0 & , \quad \hat{\mu} < 0 \end{cases}$$

where, of course, $\lambda(0) = \lambda(\mu = 0) = \dfrac{\mathcal{L}(0, \hat{\hat{\vec{\theta}}}(0))}{\mathcal{L}(\hat{\mu}, \hat{\vec{\theta}})}$

Note that the difference with the case of the test statistic t_0, also used to test $\mu = 0$, is that in that case one may reject the $\mu = 0$ hypothesis for either an upward or a downward fluctuation. This is appropriate when a (new) phenomenon could lead to an increase or decrease in the number of events found (in other word in a counting event rate).

More typically, in an invariant mass spectrum, the appearance of a resonance/peak/structure can be associated to an upward fluctuation (if bkg has to mimic this excess found in the data distribution). In the latter case assuming $\mu \geq 0$ makes sense.

When using q_0, however, we consider the data to show lack of agreement with the bkg-only hypothesis only if $\hat{\mu} > 0$. Namely a value of $\hat{\mu}$ below 0 may indeed constitute evidence against the bkg-only model but this type of discrepancy does not show that the data contain signal events, but rather would point to some systematic errors/effects. However typically we assume that systematic uncertainties are dealt with by the nuisance parameters.

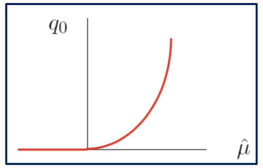
Test statistic for discovery of a positive signal - II

To quantify the level of disagreement between the data and the $\mu = 0$ hypothesis of using the observed value of q_0 …
… we compute the p-value similarly to what done previously with t_μ :

$$q_0 = \int_{q_{0,obs}}^{\infty} f(q_0|0)\, dq_0$$

$f(q_0|0)$ → p.d.f. of the statistic q_0 under assumption of bkg-only hypothesis ($\mu = 0$)

$q_{0,obs}$ ← value of the test statistic observed in the data



Assuming the validity of the **Wald approximation**, written with $\mu = 0$, and in the large limit sample ($N \to \infty$) one has:

$$q_0 = \begin{cases} -2ln\lambda(0) \cong \dfrac{\hat{\mu}^2}{\sigma^2} & , \hat{\mu} \geq 0 \\ 0 & , \hat{\mu} < 0 \end{cases}$$

… where $\hat{\mu}$ follows a Gaussian distrib. with mean μ' and standard deviation σ

For the special case of $\mu' = 0$ (= μ) … the p.d.f. reduces to the following easy form:

equal mixture

$$f(q_0|0) = \frac{1}{2}\delta(q_0) + \frac{1}{2}\frac{1}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{q_0}} \cdot e^{-\frac{1}{2}q_0}$$

delta function at 0 — 1 d.o.f. chi-square distrib. $\frac{1}{2}\chi_1^2$

called **"half chi-square" distribution**

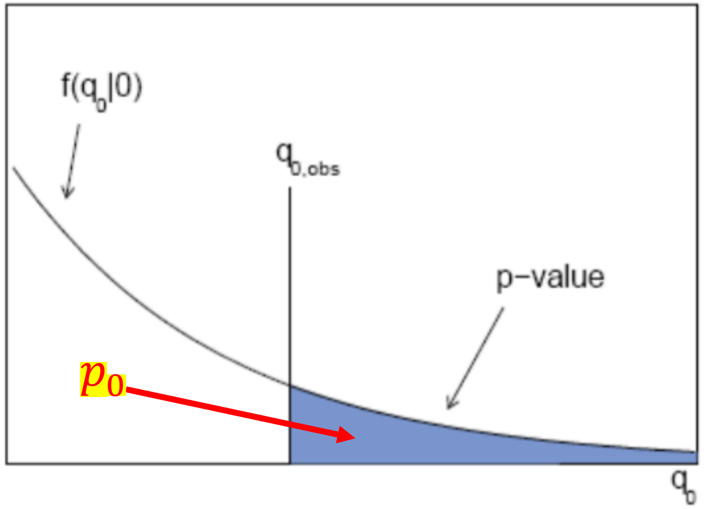
The corresponding cumulative distribution of $\boldsymbol{q_0}$ is (can be checked):

$$F(q_0|0) \cong \Phi(\sqrt{q_0})$$

Test statistic for discovery of a positive signal - III

Now the p-value of a hypothetical value of μ for an observed value of $\boldsymbol{q_o}$ is :

$$p_0 = \int_{q_{0,obs}}^{\infty} f(q_0|0)dq_0 = 1 - \int_{-\infty}^{q_{0,obs}} f(q_0|0)dq_0 = 1 - F(q_0|0) \cong 1 - \Phi(\sqrt{q_0})$$



… and the significance corresponding to this p-value is,
by rewriting the expression $Z = \Phi^{-1}(1 - p_Z)$:

$$Z_0 = \Phi^{-1}(1 - p_0) \cong \Phi^{-1}\left(1 - 1 + \Phi(\sqrt{q_0})\right) = \Phi^{-1}\,\Phi(\sqrt{q_0}) = \sqrt{q_0}$$

(note: this is eq.52 of *Cowan et al.*)

… where q_0 is the test statistic under assumption of bkg-only hypothesis ($\mu = 0$), namely $q_0 = \tilde{t}_{\mu=0}$,
with $\tilde{t}_\mu = \boldsymbol{t_\mu} = -2ln\lambda(\mu)$ under the assumption that $\mu \geq 0$.

Thus: under the assumption that $\mu \geq 0$: $q_0 = 2\left[-ln\mathcal{L}\left(\mu = 0, \hat{\hat{\theta}}\right) - \left(-ln\mathcal{L}(\hat{\mu} \geq 0, \hat{\theta})\right)\right] \equiv 2\,[NLL0 - NLL1]$ &(*) $Z_0 \cong \sqrt{q_0} = \sqrt{2(\lambda_0 - \lambda_1)}$

in the large limit sample ($N \to \infty$) & in the Wald approximation and for Wilks' theorem (assuming its application conditions hold).

With these assumptions, the *asymptotic form* of the p.d.f. of the test statistic q_0 is: $f(q_0|\mu = 0) = \frac{1}{2}\delta(q_0) + \frac{1}{2}\chi_1^2$

(*) using the simplified notation:
$\lambda_0 \equiv NLL0 = -ln$ (Likelihood for the null hypothesis/bkg-only ($\mu = 0$))
$\lambda_1 \equiv NLL1 = -ln$ (Likelihood for the hypothesis/sig+bkg ($\mu \geq 0$))

PART-2 / Computational aspects

A.Pompili - SDAL course - Exercise 11

Computational aspects of the Statistical Significance - I

The ***(Gauss) error function*** erf is a special function (i.e. a complex function of a complex number z):

$$\mathrm{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z e^{-y^2}\, dy = \left(\int_{-z}^{z} \frac{1}{\sqrt{\pi}} e^{-y^2}\, dy \right)$$

…but in many applications the function argument z is a real number (and also the function is real).

Note that for the Gaussian $\mathrm{G}(x) = \frac{1}{\sqrt{2\pi}\cdot\sigma} e^{-\frac{1}{2}\cdot\left(\frac{x-\mu}{\sigma}\right)^2}$ in the case for which $\mu = 0$ and $\sigma = \frac{1}{\sqrt{2}}$ one gets $\tilde{G}(x) = \frac{1}{\sqrt{\pi}} e^{-(x)^2}$

(which is not the standard Gaussian!)

The erf is widely used in statistical computations where it's known as the standard normal cumulative probability. It's important to clarify the link between the normal c.d.f. and the erf, since the former is often expressed in terms of the latter (I report the result here, see next slide for the explanation):

$$\Phi(z) = \int_{-\infty}^{z} G(x|0,1)dx = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{z} e^{-x^2/2} dx \Rightarrow \mathrm{erf}(z) = 2\Phi(z\sqrt{2}) - 1 \Leftrightarrow \frac{1+\mathrm{erf}(z)}{2} = \Phi(z\sqrt{2})$$

(c.d.f. of the standard Gaussian)

By definition, the Error Function is

$$\mathrm{Erf}(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2}\,\mathrm{d}t.$$

Writing $t^2 = z^2/2$ implies $t = z/\sqrt{2}$ (because t is not negative), whence $\mathrm{d}t = \mathrm{d}z/\sqrt{2}$. The endpoints $t = 0$ and $t = x$ become $z = 0$ and $z = x\sqrt{2}$. To convert the resulting integral into something that looks like a cumulative distribution function (CDF), it must be expressed in terms of integrals that have lower limits of $-\infty$, thus:

$$\mathrm{Erf}(x) = \frac{2}{\sqrt{2\pi}} \int_0^{x\sqrt{2}} e^{-z^2/2}\mathrm{d}z = 2\left(\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x\sqrt{2}} e^{-z^2/2}\mathrm{d}z - \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{0} e^{-z^2/2}\mathrm{d}z\right).$$

Those integrals on the right hand size are both values of the CDF of the standard Normal distribution,

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-z^2/2}\mathrm{d}z.$$

Specifically,

$$\mathrm{Erf}(x) = 2(\Phi(x\sqrt{2}) - \Phi(0)) = 2\left(\Phi(x\sqrt{2}) - \frac{1}{2}\right) = 2\Phi(x\sqrt{2}) - 1.$$

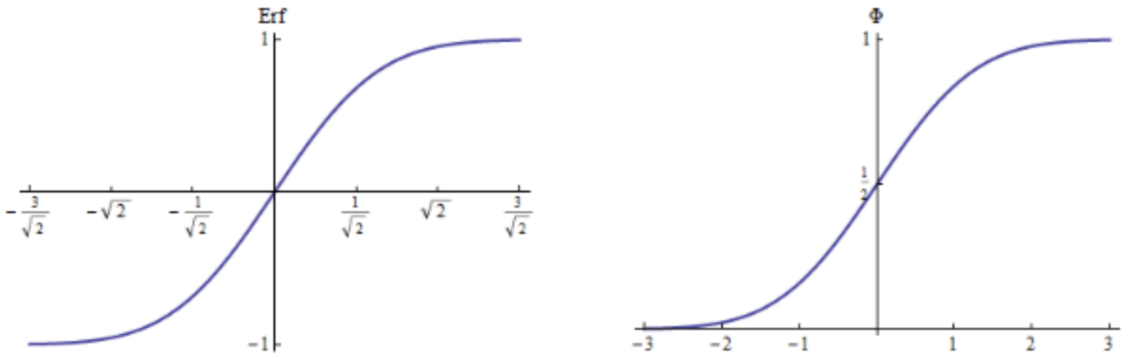
This shows how to express the Error Function in terms of the Normal CDF. Algebraic manipulation of that easily gives the Normal CDF in terms of the Error Function:

$$\Phi(x) = \frac{1 + \mathrm{Erf}(x/\sqrt{2})}{2}.$$

This relationship (for real numbers, anyway) is exhibited in plots of the two functions. The graphs are identical curves. The coordinates of the Error Function on the left are converted to the coordinates of Φ on the right by multiplying the x coordinates by $\sqrt{2}$, adding 1 to the y coordinates, and then dividing the y coordinates by 2, reflecting the relationship

$$\Phi(x\sqrt{2}) = \frac{\mathrm{Erf}(x) + 1}{2}$$

in which the notation explicitly shows these three operations of multiplication, addition, and division.



https://stats.stackexchange.com/questions/187828/how-are-the-error-function-and-standard-normal-distribution-function-related

Computational aspects of the Statistical Significance - III

It is possible - again - to extract the argument/variable:

$$\Phi(Z) = 1 - p_Z \quad \Longrightarrow \quad Z = \Phi^{-1}(1 - p_Z) \quad \Rightarrow \quad \Phi\left(Z\sqrt{2}\right) = \frac{1 + \text{erf}(Z)}{2} \quad \Longrightarrow \frac{Z}{\sqrt{2}} = \frac{1}{2}\Phi^{-1}\left(1 + \text{erf}(Z)\right) \quad \Leftrightarrow \quad Z = \frac{1}{\sqrt{2}}\Phi^{-1}\left(1 + \text{erf}(Z)\right)$$

It is also useful to introduce the ***complementary error function*** erfc : $\text{erfc}(Z) = 1 - \text{erf}(Z)$

This definition implies: $\text{erf}(Z) = 1 - \text{erfc}(Z)$ and $1 + \text{erf}(Z) = 2 - \text{erfc}(Z)$

$$Z = \frac{1}{\sqrt{2}}\Phi^{-1}\left(2 - \text{erfc}(Z)\right)$$

It can be also useful the property that the error function is an odd function: $\text{erf}(-Z) = -\text{erf}(Z)$

In the next slide I report all the mathematical expressions that work like convertion relations between $\Phi(Z)$ and $\text{erf}(Z)$ (found on the web). We wil see later that they are useful to calculate explicitely the Statistical Significances.

Relating Φ and erf

There's nothing profound here, just simple but error-prone calculations that I've done so often that I decided to save the results.

Let $\Phi_{\mu,\sigma}(x)$ be the CDF of a normal random variable with mean μ and standard deviation σ.

$$\Phi_{\mu,\sigma}(x) = \frac{1}{\sqrt{2\pi}\sigma}\int_{-\infty}^{x} \exp\left(-\frac{(t-\mu)^2}{2\sigma^2}\right) dt$$

Let $\Phi(x)$ with no subscripts be the CDF of a standard normal random variable, *i.e.* $\mu = 0$ and $\sigma = 1$. Let $\Phi_c(x) = 1 - \Phi(x)$, the complementary CDF of a standard normal.

The error function is defined as

$$\mathrm{erf}(x) = \frac{2}{\sqrt{\pi}}\int_0^x \exp(-t^2)\, dt$$

and the complementary error function is defined as

$$\mathrm{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^{\infty} \exp(-t^2)\, dt = 1 - \mathrm{erf}(x).$$

These relations below follow directly from the definitions.

$$\begin{aligned}
\Phi_{\mu,\sigma}(x) &= \Phi\left(\frac{x-\mu}{\sigma}\right) \\
\Phi(x) &= \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right) \\
\mathrm{erf}(x) &= 2\Phi(\sqrt{2}x) - 1 \\
\Phi^{-1}(x) &= \sqrt{2}\,\mathrm{erf}^{-1}(2x-1) \\
\mathrm{erf}^{-1}(x) &= \frac{1}{\sqrt{2}}\Phi^{-1}\left(\frac{x+1}{2}\right) \\
\Phi_{\mu,\sigma}(x) &= \frac{1}{2}\left(1 + \mathrm{erf}\left(\frac{x-\mu}{\sqrt{2}\sigma}\right)\right) \\
\Phi^{-1}_{\mu,\sigma}(x) &= \sqrt{2}\sigma\,\mathrm{erf}^{-1}(2x-1) + \mu \\
\Phi_c(x) &= \frac{1}{2}\mathrm{erfc}\left(\frac{x}{\sqrt{2}}\right) \\
\Phi_c^{-1}(x) &= \Phi^{-1}(1-x) \\
\Phi_c^{-1}(x) &= \sqrt{2}\,\mathrm{erfc}^{-1}(2x) \\
\mathrm{erfc}(x) &= 2\Phi_c(\sqrt{2}x) \\
\mathrm{erfc}^{-1}(x) &= \frac{1}{\sqrt{2}}\Phi_c^{-1}\left(\frac{x}{2}\right)
\end{aligned}$$

(the expressions pointed out with arrows are those that we use)

PART-3 / Estimation implemented at code level

A.Pompili - SDAL course - Exercise 11

How to execute the code

The code is in the macro **`myGenExpGausStatSignif.C`** and can be executed in the following way

```
[pompili@pompilic7 exp-gauss]$ root -l
root [0] .L myGenExpGausStatSignif.C++
Info in <TUnixSystem::ACLiC>: creating shared library /home/pompili/SDAL-2022/Esercitazione-11/exp-gauss/./myGenExpGausStatSignif_C.so
root [1] myGenExpGausStatSignif("100000",100)
Events = 100000
sigCand =500
bkgCand =99500

------
msRand = 1671477488657
------
```

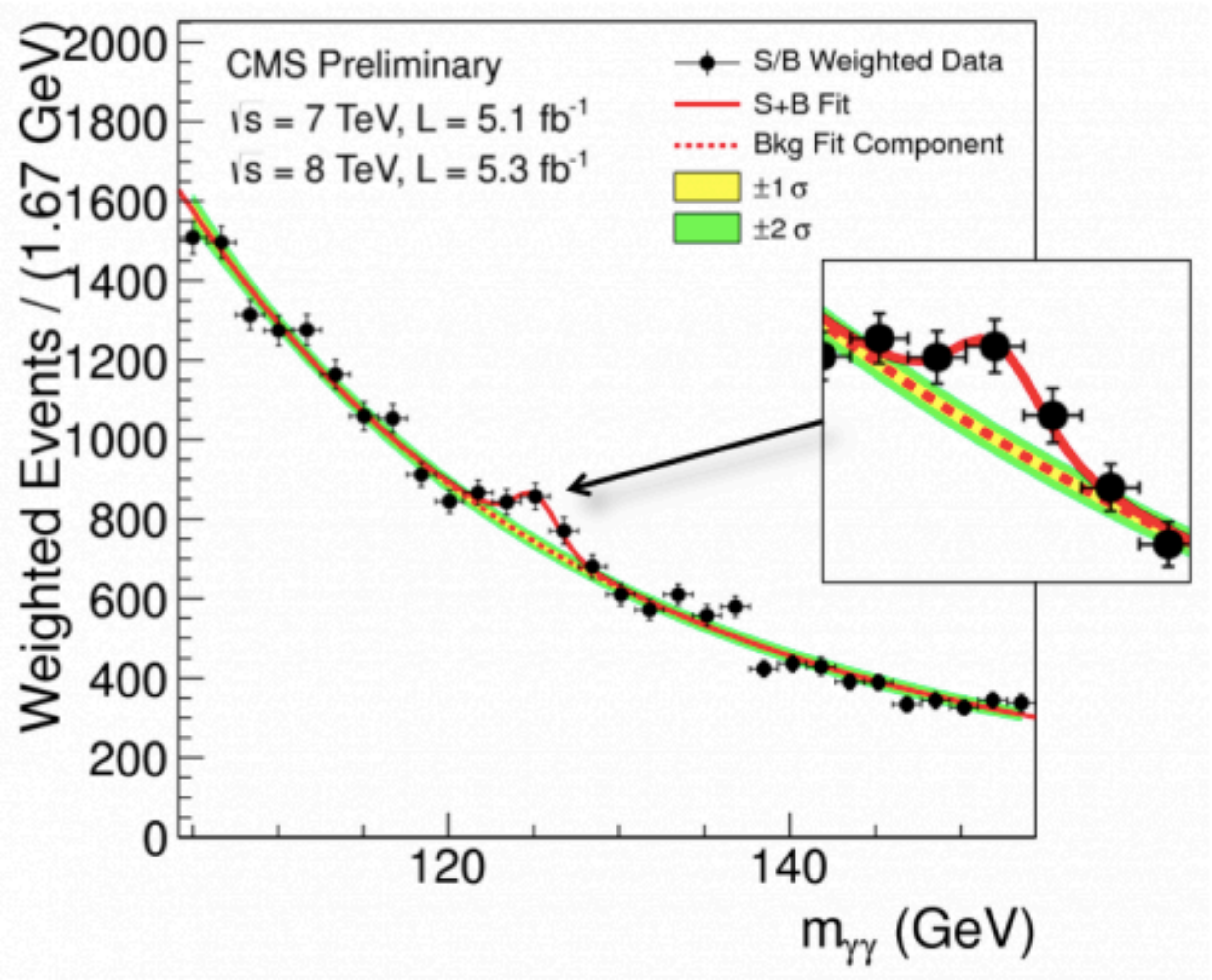
# of bins needed to display the data generated unbinned

# of generated events

the fraction of signal events is decided (hardwired) in the code;
default is 5-per-mill so to get a statistical significance for the signal of about 5σ

What kind of generated data?

The generation model/shape can remind to someone something already seen “around” :



What the code does? - I

Configuration & data generation initial part:

```
void myGenExpGausStatSignif(TString argv, int bins){
   //
   gROOT->SetStyle("Plain");
   gStyle->SetOptStat(10);
   gStyle->SetOptFit(111);
   //
   int events = atoi(argv.Data());  // converte string "numero" in numero intero
   TString name = argv;
   //
   RooRealVar xvar("xvar", "", 18., 34.);
   xvar.setBins(bins);
   //
   //-- BKG MODEL
   //
   RooRealVar m0("m0","m0",-0.1, -2., 2.);
   RooExponential myExp("myExp","Exponential", xvar, m0);
   //
   //-- SIGNAL MODEL
   //
   RooRealVar meanG("meanG","Gaussian mean",26., 25., 27.);
   RooRealVar sigmaG("sigmaG","Gaussian sigma/resolution",0.4,0.36,0.44);
   RooGaussian myGauss("myGauss","Gaussian",xvar,meanG,sigmaG);
   //
   //-- TOTAL MODEL
   //
   cout << "Events = " << events << endl;
   //
   //---suppose a signal rapresented by the 5-per-mill of the whole distribution:
   double sigFrac = 0.005;
   //
   int sigCand = sigFrac * events;
   //int sigCandM = 0.1 * sigFrac * events;
   //int sigCandM =0;
   int sigCandP = 5 * sigFrac * events;
   cout << "sigCand =" << sigCand << endl;
   //
   int bkgCand = (1 - sigFrac) * events;
   int bkgCandM = 0.1 * (1 - sigFrac) * events;
   int bkgCandP = 5 * (1 - sigFrac) * events;
   cout << "bkgCand =" << bkgCand << endl;
   //
   // note that signal yield is positive by definition; generated value is given by sigCand:
   RooRealVar yield_sig("yield_sig","yield of Gaussan signal component", sigCand, 0, sigCandP);
   RooRealVar yield_bkg("yield_bkg","yield of Exponential bkg component", bkgCand, bkgCandM, bkgCandP);
   RooAddPdf total("totalPDF", "totalPDF", RooArgList(myGauss,myExp), RooArgList(yield_sig,yield_bkg));
   //
   //--> Generating pseudo-data
   ///////////////////////////
   //
   timeval trand;
   gettimeofday(&trand,NULL);
   long int msRand = trand.tv_sec * 1000 + trand.tv_usec / 1000;
   cout << "\n------" << endl;
   cout << "msRand = " << msRand;
   cout << "\n------" << endl;
   RooRandom::randomGenerator()->SetSeed(msRand);
   //
   RooDataSet* data = total.generate(xvar,events);
   //TH1D* histo_data = (TH1D*)data->createHistogram("histo_data",xvar,Binning(bins,xvar.getMin(),xvar.getMax()));
   //
   cout << "=================================================================" << endl;
```

Exponential model for the bkg

Gaussian model for the signal

generated signal fraction = 0.5%

the *strength* parameter (signal yield) $\mu \geq 0$ →

generation →

→ no need for the histogram

What the code does? - II

I overall generate 100K events with 500 signal events
[gaussian model with: width = known "mass" resolution (0.4) and known "mass" (26);
background is exponential and its parameters are the *nuisance* ones].

After the first part of generation, the code performs a **sequence of 3 fits**:

1) fully free fit Sig+Bkg (signal modelled with a full free Gaussian)
2) constrained fit Sig+Bkg (signal modelled with a Gaussian with mass and width settled; signal yield free)
3) bkg-only fit (signal yield constrained to 0)

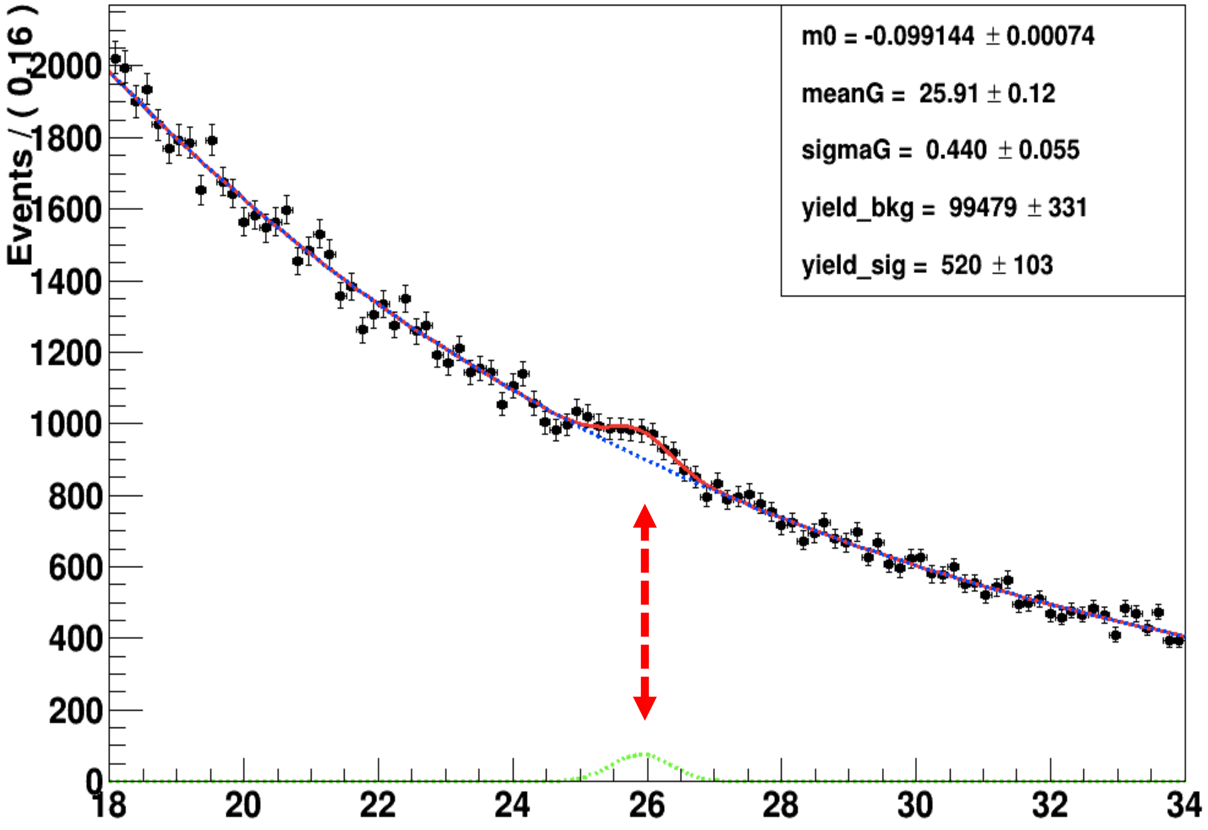
When estimating the *local* statistical significance of the signal, the signal yield will be the only free parameter of interest:
It will be free (but costrained to be positive) in Fit-2 and set to 0 in Fit-3.
Since it is the *local* SS to be estimated, it is indeed possible to …

- **set the gaussian width to the resolution that typically one knows from simulation studies, and …**
- **set the mass to the nominal (generated) mass since the resonance/particle/state is assumed to be known**

What the code does? - III

Fit-1: S+B fully free

```
cout << "=================================================================" << endl;
cout << "FULLY FREE FIT : Gaussian model for the fit without constraints " << endl;
cout << "-----------------------------------------------------------------" << endl;
//
//--> Fitting pseudo-data with total model
//
// Note that fit is EXTENDED without having to say it explicitely
//
RooAbsReal* nll_free = total.createNLL(*data,NumCPU(4));
//RooMinuit min_free(*nll_free); // changing a bit with previous exercises: using Minuit2 now.
RooMinimizer min_free(*nll_free);
min_free.setMinimizerType("Minuit2");
min_free.migrad();
min_free.hesse();
RooFitResult* fitres_free = min_free.save();
//
TCanvas *myC = new TCanvas("RooCanvas","Roofit Canvas", 1100, 750);
myC->cd();
//
RooPlot* xframe_free = xvar.frame("");
xframe_free->SetTitle("");
data->plotOn(xframe_free);
total.plotOn(xframe_free,LineColor(kRed));
total.plotOn(xframe_free,Components(RooArgSet(myGauss)),LineColor(kGreen),LineStyle(kDashed));
total.plotOn(xframe_free,Components(RooArgSet(myExp)),LineColor(kBlue),LineStyle(kDashed));
//data->plotOn(xframe_free);
total.paramOn(xframe_free, Layout(0.6,0.9,0.9));
xframe_free->getAttText()->SetTextSize(0.03);
xframe_free->Draw();
//
myC->SaveAs("myFull_Free_Fit.png");
myC->Update();
myC->Clear();
//
```



What the code does? - IV

Fits 2 & 3 part:

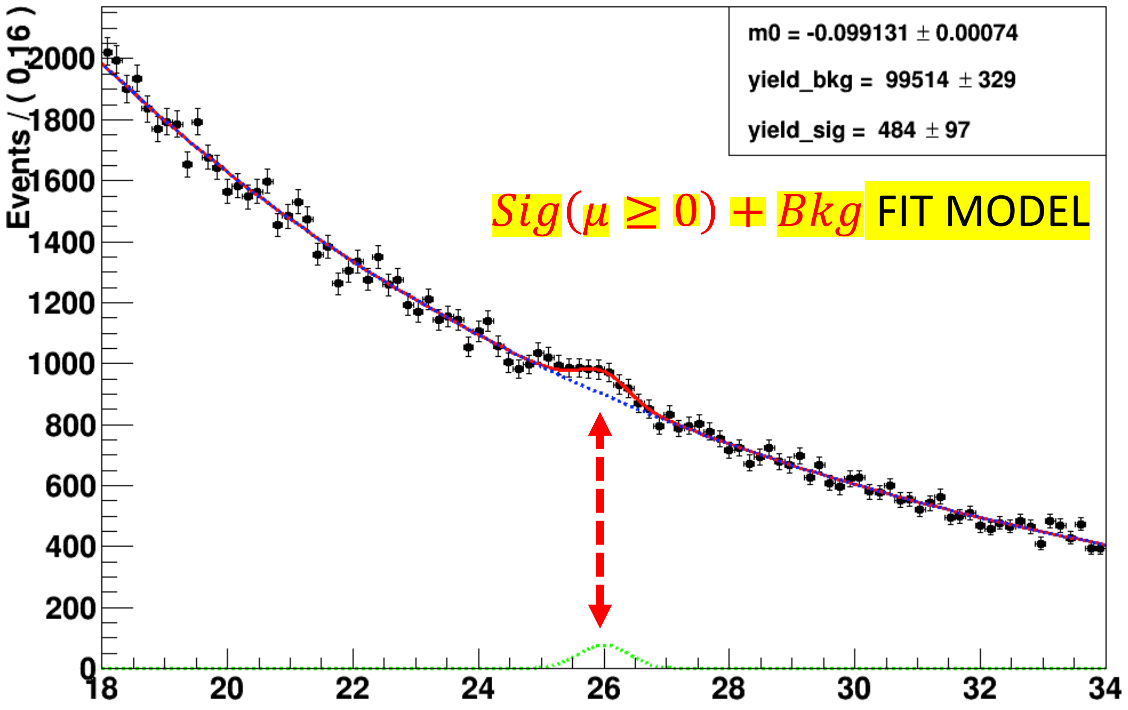
```
cout << "==================================================================================================" << endl;
cout << "FULL FIT with CONSTRAINED mass & width to their (knwon/generated) values, to apply Wilks Theorem " << endl;
cout << "--------------------------------------------------------------------------------------------------" << endl;
//
//---- when calculating statistical significance applying Wilks theorem decomment next 4 lines:
//
meanG.setVal(26.); // otherwise it sets to constant with the best value from previous fit.
meanG.setConstant(kTRUE);
cout << "Setting the mass to the nominal/generated mass " << meanG.getValV() << endl;
sigmaG.setVal(0.4); // otherwise it sets to constant with the best value from previous fit.
sigmaG.setConstant(kTRUE);
cout << "Setting the Gaussian width to the mass resolution " << sigmaG.getValV() << endl;
//
RooAbsReal* nll = total.createNLL(*data,NumCPU(4));
//RooMinuit min(*nll); // changing a bit with previous exercises: using Minuit2
RooMinimizer min(*nll);
min.setMinimizerType("Minuit2");
min.migrad();
min.hesse();
RooFitResult* fitres = min.save();
//fitres->Print("v");
Double_t min_NLL_total = fitres->minNll();
cout << "-log(L) at minimum : value for total (sig+bkg) model :" << min_NLL_total << endl;
//
myC->cd();
RooPlot* xframe = xvar.frame("");
xframe->SetTitle("");
data->plotOn(xframe);
total.plotOn(xframe,LineColor(kRed));
total.plotOn(xframe,Components(RooArgSet(myGauss)),LineColor(kGreen),LineStyle(kDashed));
total.plotOn(xframe,Components(RooArgSet(myExp)),LineColor(kBlue),LineStyle(kDashed));
//data->plotOn(xframe);
total.paramOn(xframe, Layout(0.6,0.9,0.9));
xframe->getAttText()->SetTextSize(0.03);
xframe->Draw();
//
myC->SaveAs("myFull_1Ndof_Fit.png");
myC->Update();
myC->Clear();
//
```

Fit-2: S+B (S constrained)

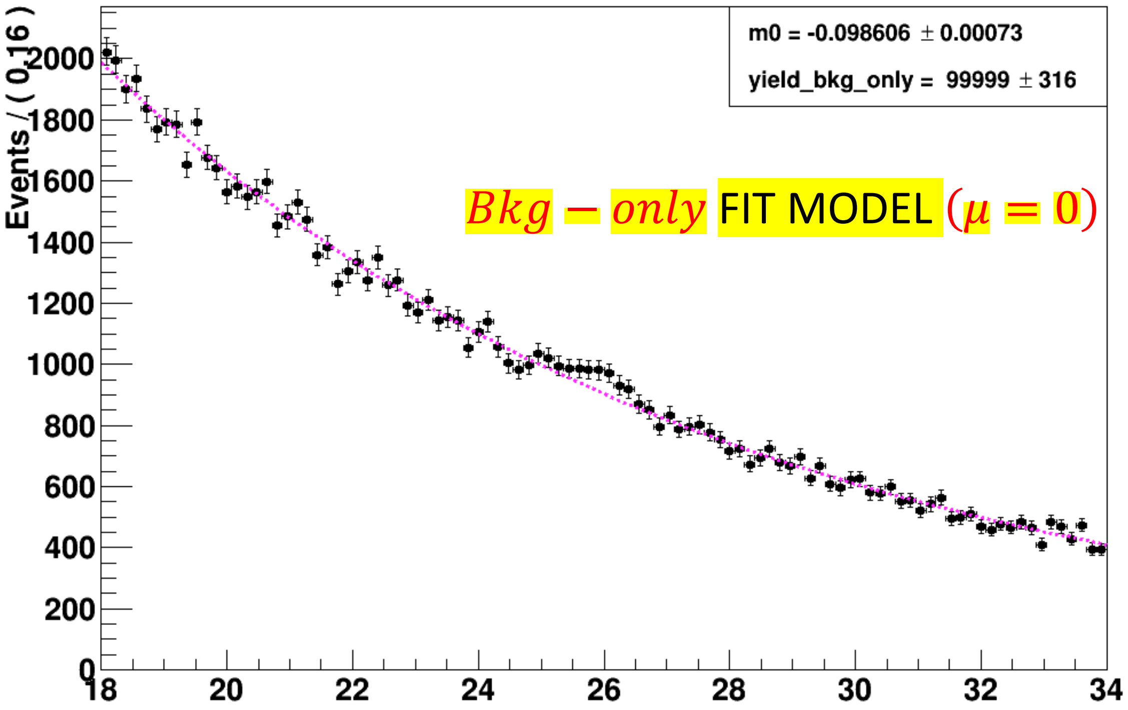
Fit-3: B-only (S=0)

```
cout << "==================================================================================================" << endl;
cout << "FIT with BACKGROUND-ONLY model " << endl;
cout << "--------------------------------------------------------------------------------------------------" << endl;
//
cout << "yield_sig from previous constrained fit = " << yield_sig.getValV() << endl;
yield_sig.setVal(0.00);
yield_sig.setConstant(kTRUE); //it sets the gaussian area to 0!
cout << "Setting the Gaussian area to " << yield_sig.getValV() << endl;
//
cout << "gaussian sigma from previous fit (check that it should be 0.4) :" << sigmaG.getVal() <<endl;
//
RooAbsReal* nll_bkg_only = total.createNLL(*data,NumCPU(4));
RooMinimizer min_bkg_only(*nll_bkg_only);
min_bkg_only.setMinimizerType("Minuit2");
min_bkg_only.migrad();
min_bkg_only.hesse();
RooFitResult* fitres_bkg_only = min_bkg_only.save();
//
Double_t min_NLL_bkgOnly = fitres_bkg_only->minNll();
cout << "-log(L) at minimum : value for bkg-only model (alternative i.e setting ZERO-AREA) : " << min_NLL_bkgOnly << endl;
//
myC->cd();
//
RooPlot* xframe_only_alt = xvar.frame("");
xframe_only_alt->SetTitle("");
data->plotOn(xframe_only_alt);
total.plotOn(xframe_only_alt,LineColor(kBlue));
total.paramOn(xframe_only_alt, Layout(0.6,0.9,0.9));
xframe_only_alt->getAttText()->SetTextSize(0.02);
xframe_only_alt->Draw();
//
myC->SaveAs("myBkgOnly_Fit.png");
myC->Update();
myC->Clear();
//
myC->cd();
//
```

The results from Fits 2&3 are:



Fit-2: S+B (S constrained)



Fit-3: B-only

What the code does? - VI

Finally the code provides, using the information of Fit-2 & Fit-3, 3 different ways to calculate the **local statistical significance** [“methods” 1,2 & 3]:

```
   //
   cout << "============================== MINIMUM NLL for each FIT ======================================" << endl;
   cout << "-log(L) at minimum : MASS and WIDTH FIXED (i.e. 1 D.O.F.) :" << min_NLL_total << endl;
   cout << "-log(L) at minimum : MASS and WIDTH FIXED and ZERO-AREA SET (i.e. BKG-ONLY) : " << min_NLL_bkgOnly << endl;
   cout << "----------------------------------------------------------------------------------------------" << endl;
   //
   cout << "lambda_0 : " << min_NLL_bkgOnly << endl;
   cout << "------------------------------" << endl;
   cout << "lambda_1 : " << min_NLL_total << endl;
   cout << "------------------------------" << endl;
   cout << "q0 = 2*(lambda_0 - lambda_1) : " << 2.*(min_NLL_bkgOnly - min_NLL_total) << endl;
   cout << "------------------------------" << endl;
   //
   cout << "============================== STAT. SIGNIF. Method-1 (STA-SIGNIF-1) ======================================" << endl;
   //
   cout << "STAT. SIGNIF. 1st method = Z0 = sqrt(q0) i.e. eq.(52) by Cowan et al. (EPJC,2011): " << sqrt(2.*(min_NLL_bkgOnly - min_NLL_total)) << endl;
   //
   cout << "-----------------------------------------------------------------------------------------------------------" << endl;
   //
   cout << "2*(p0-value) = TMath::Prob((2.*(min_NLL_bkgOnly - min_NLL_total)),1) = " << TMath::Prob((2.*(min_NLL_bkgOnly - min_NLL_total)),1) << endl;
   //
   cout << "1 - erf (sqrt(q0/2)) : " << 1. - TMath::Erf(sqrt(2.*(min_NLL_bkgOnly - min_NLL_total)/2.)) << " = TMath::Prob((2.*(min_NLL_bkgOnly - min_NLL_total)),1)" << endl;
   //
   cout << "p-value = 1/2*[1 - erf (sqrt(q0/2))] = " << 0.5*(1. - TMath::Erf(sqrt(2.*(min_NLL_bkgOnly - min_NLL_total)/2.))) << endl;
   //
   cout << "============================== STAT. SIGNIF. Method-2 (STA-SIGNIF-2) ======================================" << endl;
   //
   Double_t stat_signif_2 = sqrt(2.)*TMath::ErfInverse(1.-(TMath::Prob((2.*(min_NLL_bkgOnly - min_NLL_total)),1)));
   cout << "STAT. SIGNIF. 2nd method = " << stat_signif_2  << " = Z0 = sqrt(2)*TMath::ErfInverse(1.-TMath::Prob(2*(min_NLL_bkgOnly - min_NLL_total),1)) : " << sqrt(2.)*TMath::ErfInverse(1.-(TMath::Pro
b(2.*(min_NLL_bkgOnly - min_NLL_total),1))) << endl;
   //
   cout << "-----------------------------------------------------------------------------------------------------------" << endl;
   //
   cout << "============================== STAT. SIGNIF. Method-3 (STA-SIGNIF-3) ======================================" << endl;
   Double_t stat_signif_3 = sqrt(2.)*TMath::ErfcInverse(TMath::Prob(2.*(min_NLL_bkgOnly - min_NLL_total),1));
   cout << "STAT. SIGNIF. 3rd method = " << stat_signif_3 << " = Z0 = sqrt(2)*TMath::ErfcInverse(TMath::Prob(2*(min_NLL_bkgOnly_alt - min_NLL_total),1)) : " << sqrt(2.)*TMath::ErfcInverse(TMath::Prob(
2.*(min_NLL_bkgOnly - min_NLL_total),1)) << endl;
   //
   cout << "-----------------------------------------------------------------------------------------------------------" << endl;
   //
   ////////////////////////////
   //
   if (myC)
     {
      myC->Close();
      delete myC;
     }
   //
}
```

Local statistical significance estimation

The output of the last part is below and provides the numerical example:

```
====== RECAP ======
lambda_0 : -783800
lambda_1 : -783813
Delta lambda = lambda_0 - lambda_1 : 12.9651
q0 = 2*(lambda_0 - lambda_1) : 25.9301
Z0 = sqrt(q0) i.e. eq.(52) by Cowan et al. : 5.09216
1 - erf (sqrt(q0/2)) : 3.54005e-07
TMath::Prob(q0,1) : 3.54005e-07
Z0 = sqrt(2.)*TMath::ErfInverse(1.-TMath::Prob(2*(min_NLL_bkgOnly_alt - min_NLL_total),1)) : 5.09216
Z0 = sqrt(2.)*TMath::ErfcInverse(TMath::Prob(2*(min_NLL_bkgOnly_alt - min_NLL_total),1)) : 5.09216
```

"method"-1 (Z0 = sqrt(q0))

just a check (1 - erf and TMath::Prob lines)

"method"-2 (ErfInverse)

"method"-3 (ErfcInverse)

Note : of course the 3 ways to estimate the stat. signif. provide the same numerical result, as it should be expected.

Advice: - the 1st way is then preferred because it's the easiest;

- the 3rd is to be preferred to the 2nd because it performs better numerically when significances are large ($Z_0 \gg 5\sigma$).

The implementation of the 3 "methods" are illustrated in the following slides!

Local statistical significance estimation: "method"-1

The first way to estimate it is simply the Cowan (et al.)'s "formula" [eq. (52) of their paper]: $Z_0 = \sqrt{q_0}$

To recap about the asymptotic formula:

Under the assumption that $\mu \geq 0$: $q_0 = 2\left[-ln\mathcal{L}\left(\mu = 0, \hat{\hat{\theta}}\right) - \left(-ln\mathcal{L}(\hat{\mu} \geq 0, \hat{\theta})\right)\right] \equiv 2\,[NLL0 - NLL1]$

In the large limit sample ($N \to \infty$) & in the Wald approximation and for Wilks' theorem (assuming its applicability conditions hold)

the *asymptotic form* of the p.d.f. of the test statistic q_0 is: $f(q_0|\mu = 0) = \frac{1}{2}\delta(q_0) + \frac{1}{2}\,\frac{1}{\sqrt{2\pi}} \cdot \frac{1}{\sqrt{q_0}} \cdot e^{-\frac{1}{2}q_0} = \frac{1}{2}\delta(q_0) + \frac{1}{2}\chi_1^2$

The corresponding cumulative distribution of q_0 is : $F(q_0|0) \cong \Phi(\sqrt{q_0})$

Therefore:

$$p_0 = \int_{q_{0,obs}}^{\infty} f(q_0|0)dq_0 = 1 - \int_{-\infty}^{q_{0,obs}} f(q_0|0)dq_0 = 1 - F(q_0|0) \cong 1 - \Phi(\sqrt{q_0})$$

... and : $Z_0 = \Phi^{-1}(1 - p_0) \cong \Phi^{-1}\left(1 - 1 + \Phi(\sqrt{q_0})\right) = \Phi^{-1}\,\Phi(\sqrt{q_0}) = \sqrt{q_0}$

In the code:

```
stat_signif = sqrt( 2.*(min_NLL_bkgOnly - min_NLL_total) ) ;
```

$q_{0,obs}$

Local statistical significance estimation: introduction to “methods”-2&3 - I

In both ways (2&3) the p-value is **explicitely calculated** and for this we need to use the function **TMath::Prob(** $q_{0,obs}$ **,1)**:

```
Double_t TMath::Prob(Double_t chi2,Int_t ndf)
{
   // Computation of the probability for a certain Chi-squared (chi2)
   // and number of degrees of freedom (ndf).
   //
   // Calculations are based on the incomplete gamma function P(a,x),
   // where a=ndf/2 and x=chi2/2.
   //
   // P(a,x) represents the probability that the observed Chi-squared
   // for a correct model should be less than the value chi2.
   //
   // The returned probability corresponds to 1-P(a,x),
   // which denotes the probability that an observed Chi-squared exceeds
   // the value chi2 by chance, even for a correct model.
   //
   //--- NvE 14-nov-1998 UU-SAP Utrecht

   if (ndf <= 0) return 0; // Set CL to zero in case ndf<=0

   if (chi2 <= 0) {
      if (chi2 < 0) return 0;
      else          return 1;
   }

   if (ndf==1) {
      Double_t v = 1.-Erf(Sqrt(chi2)/Sqrt(2.));
      return v;
   }

   // Gaussian approximation for large ndf
   Double_t q = Sqrt(2*chi2)-Sqrt(Double_t(2*ndf-1));
   if (ndf > 30 && q > 5) {
      Double_t v = 0.5*(1-Erf(q/Sqrt(2.)));
      return v;
   }

   // Evaluate the incomplete gamma function
   return (1-Gamma(0.5*ndf,0.5*chi2));
}
```

$$= 1 - \mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)$$

TMath::Prob(q_0 **,1)** (with n.d.f.=1):

Local statistical significance estimation: introduction to "methods"-2&3 - II

Since… $p_0 = \int_{q_{0,obs}}^{\infty} f(q_0|0)dq_0 = 1 - \int_{-\infty}^{q_{0,obs}} f(q_0|0)dq_0 = 1 - F(q_0|0) \cong 1 - \Phi(\sqrt{q_0})$ (discussed in slide 9)

One of the expressions relating the Φ with the erf is: $\Phi(x) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{x}{\sqrt{2}}\right)\right]$ $\xrightarrow{x=\sqrt{q_0}}$ $\Phi(\sqrt{q_0}) = \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{\sqrt{q_0}}{\sqrt{2}}\right)\right]$

Thus: $p_0 \cong 1 - \Phi(\sqrt{q_0}) = 1 - \frac{1}{2}\left[1 + \mathrm{erf}\left(\frac{\sqrt{q_0}}{\sqrt{2}}\right)\right] = 1 - \frac{1}{2} - \frac{1}{2}\mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right) = \frac{1}{2}\left[1 - \mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)\right]$

$\left[1 - \mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)\right] \cong 2p_0$ → TMath::Prob(q_0 ,1)

"Method"-2 & : "Method"-3 are two **equivalent** ways to …
… calculate **explicitely** the local statistical significance starting from: $Z_0 = \Phi^{-1}(1 - p_0)$ …where… $p_0 \cong \frac{1}{2}\left[1 - \mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)\right]$

(as discussed in next slides)

Local statistical significance estimation: "method"-2

By using… $\Phi^{-1}(x) = \sqrt{2} \cdot \mathrm{erf}^{-1}(2x-1)$ (one of the "conversion" formulas in slide 14)

$x \equiv (1 - p_0)$

"Method"-2 : $Z_0 = \Phi^{-1}(1-p_0) = \sqrt{2} \cdot \mathrm{erf}^{-1}(2(1-p_0)-1) = \sqrt{2} \cdot \mathrm{erf}^{-1}(1-2p_0)$ …where… $2p_0 \cong \left[1 - \mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)\right]$

to be implemented in the code as :

```
stat_signif = sqrt(2.)*TMath::ErfInverse(1. - TMath::Prob( 2.*(min_NLL_bkgOnly - min_NLL_total) , 1) )
```

$2p_0$ $q_{0,obs}$

Local statistical significance estimation: "method"-3

By using… $\Phi^{-1}\left(1-\frac{x}{2}\right) = \Phi_c^{-1}\left(\frac{x}{2}\right) = \sqrt{2}\cdot \mathrm{erfc}^{-1}(x)$ (one of the "conversion" formulas in slide 14)

$x = (2p_0)$

"Method"-3 : $Z_0 = \Phi^{-1}(1-p_0) = \sqrt{2}\cdot\ \mathrm{erfc}^{-1}\ (2p_0)$ …where… $2p_0 \cong \left[1-\mathrm{erf}\left(\sqrt{\frac{q_0}{2}}\right)\right]$

to be implemented in the code as :

```
stat_signif = sqrt(2.)*TMath::ErfcInverse(TMath::Prob(2.*(min_NLL_bkgOnly - min_NLL_total),1))
```

Reverting the logic it can be shown that the two methods are equivalent starting from their final expression:

$$Z_0 = \Phi^{-1}(1-p_0) = \sqrt{2}\cdot\ \mathrm{erfc}^{-1}\ (2p_0) = \sqrt{2}\cdot\ \mathrm{erf}^{-1}\ (1-2p_0) = \Phi^{-1}(1-p_0) = Z_0$$

$$\mathrm{erf}^{-1}(x) = \frac{1}{\sqrt{2}}\Phi^{-1}\left(\frac{1}{2}+\frac{x}{2}\right)$$

$$\mathrm{erfc}^{-1}(y) = \frac{1}{\sqrt{2}}\Phi^{-1}\left(1-\frac{y}{2}\right)$$

$y = 1 - x$

$$\mathrm{erfc}^{-1}(1-x) = \frac{1}{\sqrt{2}}\Phi^{-1}\left(1-\frac{1-x}{2}\right) = \frac{1}{\sqrt{2}}\Phi^{-1}\left(\frac{1}{2}+\frac{x}{2}\right) = \mathrm{erf}^{-1}(x)$$